```
#include<lpc214x.h>
#include <stdio.h>

#define LCD_PORT 0x00FF0000
#define EN 1<<10
#define RS 1<<11
#define RW 1<<20
#define LCD_SHIFT 16


void lcd_delay(unsigned int time)
{
        int i,j;
        for(i=0;i<time;i++)
                for(j=0;j<200;j++);
}


void LCD_data(unsigned char ch)
{
        IOCLR1 = LCD_PORT;
        IOSET1 = ch<<LCD_SHIFT;
        IOSET0 = RS;
        IOCLR0 = RW;
        IOSET0 = EN;
        lcd_delay(100);
        IOCLR0 = EN;
        lcd_delay(100);
}

void LCD_cmd(unsigned char ch)
```

```
{
        IOCLR1 = LCD_PORT;
        IOSET1 = ch<<LCD_SHIFT;
        IOCLR0 = RS;
        IOCLR0 = RW;
        IOSET0 = EN;
        lcd_delay(100);
        IOCLR0 = EN;
        lcd_delay(100);
}

void LCD_init(void)
{
        PINSEL0 &= 0xFF0FFFFF;
        PINSEL1 &= 0xFFFFFCFF;
        PINSEL2 &= 0xFFFFFFF3;
        IODIR0 = RS | EN | RW;
        IODIR1 = LCD_PORT;
        LCD_cmd(0x38);
        LCD_cmd(0x06);
        LCD_cmd(0x0C);
        LCD_cmd(0x01);
        LCD_cmd(0x80);
}

void LCD_display(int row, int pos, char *ch)
{
        unsigned char temp;
        if(row==1)
        {
                temp = 0x80 | (pos-1);
        }
        else
        {
                temp = 0xC0 | (pos-1);
        }
        LCD_cmd(temp);
        while(*ch)
                LCD_data(*ch++);
}


void ADCInit(void)
{
                PINSEL1 |= 0x05000000;


}

unsigned int ADC_Read(unsigned char channel)
{
        static unsigned int ad1_data;

        AD0CR = 0x00200300 | (1<<channel);
        AD0CR |= 1<<24;

        while(!(AD0GDR & 0x80000000));
        ad1_data = (AD0GDR & 0x0000FFC0)>>6;

        return ad1_data;
}
```

```
void delay(unsigned int time)
{
        unsigned int i,j;
        for(i=0;i<time;i++)
                for(j=0;j<5000;j++);
}

int main()
{
        unsigned int temp;
        char buf[16];

        LCD_init();
        ADCInit();
        while(1)
        {
                temp = ADC_Read(1);
                sprintf(buf," ADC result:0x%03X",temp);
                LCD_display(1,1,buf);
                delay(20);
        }
}
```

```
#include<lpc214x.h>
```

```
#include <stdio.h>


#define LCD_PORT 0x00FF0000

#define EN 1<<10

#define RS 1<<11

#define RW 1<<20

#define LCD_SHIFT 16



void lcd_delay(unsigned int time)

{

        int i,j;

        for(i=0;i<time;i++)

                for(j=0;j<200;j++);

}



void LCD_data(unsigned char ch)

{

        IOCLR1 = LCD_PORT;

        IOSET1 = ch<<LCD_SHIFT;

        IOSET0 = RS;

        IOCLR0 = RW;

        IOSET0 = EN;

        lcd_delay(100);

        IOCLR0 = EN;

        lcd_delay(100);

}


void LCD_cmd(unsigned char ch)

{
        IOCLR1 = LCD_PORT;
        IOSET1 = ch<<LCD_SHIFT;
        IOCLR0 = RS;
        IOCLR0 = RW;
        IOSET0 = EN;
        lcd_delay(100);
        IOCLR0 = EN;
        lcd_delay(100);
}

void LCD_init(void)
{
        PINSEL0 &= 0xFF0FFFFF;
        PINSEL1 &= 0xFFFFFCFF;
        PINSEL2 &= 0xFFFFFFF3;
        IODIR0 = RS | EN | RW;
        IODIR1 = LCD_PORT;
        LCD_cmd(0x38);
        LCD_cmd(0x06);
        LCD_cmd(0x0C);
        LCD_cmd(0x01);
        LCD_cmd(0x80);
}

void LCD_display(int row, int pos, char *ch)
{
        unsigned char temp;
        if(row==1)
        {
                temp = 0x80 | (pos-1);
        }
```

```
    else
    {
        temp = 0xC0 | (pos-1);
    }
    LCD_cmd(temp);
    while(*ch)
        LCD_data(*ch++);
}


void ADCInit(void)
{
        PINSEL1 |= 0x05000000;


}

unsigned int ADC_Read(unsigned char channel)
{
    static unsigned int ad1_data;

    AD0CR = 0x00200300 | (1<<channel);
    AD0CR |= 1<<24;

    while(!(AD0GDR & 0x80000000));
    ad1_data = (AD0GDR & 0x0000FFC0)>>6;

    return ad1_data;
}
```

```
void ADC_ISR() _irq
{
        unsigned int temp;
        char buf[16];
        temp = ADC_Read(1);
        sprintf(buf," ADC result:0x%03X",temp);
        LCD_display(1,1,buf);
        delay(20);
        AD0INTEN = 0;
        VICVectAddr=0;
}



int main(void)
{
        LCD_init();
        ADCInit();

        AD0INTEN = 0x00000002;
        VICVectAddr0 = (unsigned int)ADC_ISR;
        VICVectCntl0 = 0x20 | 18;
        VICIntEnable = 1 << 18;
        delay(20);

        while(1);
        return 0;
}
```